UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/815,038	03/31/2004	Viktor Benz	P04,0110	7828

7590 04/08/2008

SCHIFF HARDIN LLP
Patent Department
6600 Sears Tower
233 South Wacker Drive
Chicago, IL 60606

EXAMINER	
MCLEAN, NEIL R	
ART UNIT	PAPER NUMBER
2625	

MAIL DATE	DELIVERY MODE
04/08/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/815,038	BENZ ET AL.	
	Examiner	Art Unit	
	Neil R. McLean	2625	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *31 March 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *21-42* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *21-42* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 July 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *3/31/2004*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Priority

1. Receipt is acknowledged of papers submitted under 35 U.S.C. 119(a)-(d), which papers have been placed of record in the file.

Election/Restrictions

2. Applicant's election with traverse of in the reply filed on 2/15/2008/ is acknowledged. The traversal is on the ground(s) that independent claims 21 and 32 are generic to both Figures 1 and 2 and if these claims are found allowable, then all the claims in the case must be allowed. This is not found persuasive because independent claims 21 and 32 were not part of the original set of claims that the Examiner used to determine the Election/Restriction requirement.

Applicant selected Species I, and stated that Claims 21-29, 32-40 and 42 read on Species I. Examiner disagrees and believes that only Claims 21-28 read on Species I.

However, the Examiner's main reference reads on both Species I and II, and since the burden on the Examiner no longer exists, the Restriction/Election Requirement has been lifted.

Claim Rejections - 35 USC § 101

3 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

4. Claims 21-42 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

In claims 21-42, a "computer program" is being recited; however, computer program would reasonably be interpreted by one of ordinary skill in the art as software, per se. This subject matter is not limited to that which falls within a statutory category of invention because it is limited to a process, machine, manufacture, or a composition of matter. Software is a function descriptive material and a function descriptive material is non-statutory subject matter.

Claim Rejections - 35 USC § 102

5. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

6. Claims 21-25, and 31-36 are rejected under 35 U.S.C. 102(e) as being anticipated by Hu et al. (US 6,535,518).

1-20. (cancelled)

Regarding Claim 21: (new)

Hu et al. discloses a method to transfer data in a server system having at least a first server **(FIG. 8 is top-level diagram for the relation between the device and server and storage.)**, comprising the steps of:

providing a computer program module **(FIG. 3 describes the software structure for the preferred embodiment for the data-driven multi-processor pipelined model. The functional relationship among the software modules is described at Column 11, line 34)** supplying data from the first server **(e.g., The Network Interface Card (NIC, 701) receives data from the network as described in Column 11, lines 34-35). FIGS. 11 and 12 are flow charts for data flow from network to storage or vice-versa.)**;

providing a reading computer program module that reads the supplied data **(In one of Hu's methods e.g., Once a file read request is sent to the host, the TWIP file system does not have to wait for response. It can proceed to process the next connection. After the host acknowledges the request (registration), the TWIP file system will go back to read the file as described in Column 17, lines 13-17)**; and

selecting one of the following transmission modes

a complete storage of the data in a file occurs before the reading computer program module reads the data **(e.g., Authorized *non-real-time data transfer***

between a network interface and a storage interface described in Column 5, lines 44-45.),

a segment-by-segment storage of the data in a file occurs such that the reading computer program module already begins with the reading of a segment while the supplying computer program module is still supplying data **(e.g., The data content then is divided into segments of integral multiples of a fixed base, a process that we call "base-multiple segmentation" (BMS) technology. For example, a base of y bytes, say 2 Kbytes, is chosen, and all data streams or files are *segmented into chunks* of integral multiples of 2 Kbytes, like 2, 4, or 8 Kbytes (padding it for the last chunk if it is not an exact integral multiple of 2 Kbytes), with an upper limit of, say, 40 Kbytes (20 times y). The maximum size is chosen based-on the requirement of isochronous real-time traffic and the switching speed, such that it will still meet the tightest real-time needs while the switching element serves the largest segments as described in Column 6, lines 3-12)**, and

a direct transmission of the data between the supplying computer program module and the reading computer program module occurs without buffering **(e.g., Authorized *real-time data transfer* between a network interface and a storage interface as described in Column 5, lines 42-43.)**

Regarding Claim 22: (new)

Hu et al. discloses a method according to claim 21 wherein the transmission mode is selected controlled by parameters, and wherein the reading computer program

module and the computer program module supplying the data cooperate via the control parameters **(e.g., Once the nature of the traffic is determined, by consulting the Expanded Routing Table (ERT) (with more information than a regular routing table), as shown in FIG. 14, a proper switching path can be selected to forward the traffic with proper QoS measurement. For instance, higher priority traffic can be given more bandwidth and/or lower delay. The forwarded traffic to the network will then be processed with the proper protocol format conversion for transmission with all the necessary error checking and/or correction as described in Column 6, lines 19-27.)**

Regarding Claim 23: (new)

Hu et al. discloses a method according to claim 21 wherein in the transmission mode with the direct transmission of the data, the reading computer program module reacts, controlled by parameters, in one of the following manners when data to be read no longer exists:

the read event is continuously repeated until data to be read is present, or until the reading computer program module receives a notification that data is no longer being supplied, or the read event is aborted **(e.g., In the second method, the host write request is intercepted by the TWIP host device driver. The TWIP host device driver then generates a write request (w_req). Then TWIP completes all outstanding read requests and sends back a write acknowledgement (w_ack) to the host and routes all future read requests to the host. Upon receiving the signal**

w_ack at the host, the TWIP host device driver releases the hold on the original write requests and proceeds to write (thick vertical line on host in FIG. 5). Once the host finishes all outstanding write operations, the TWIP device driver detects this and sends write-release (w_rel) to TWIP. When TWIP receives w_rel, it resumes the bypass function if it can handle the new incoming requests as described in Column 17, lines 17-28).

Regarding Claim 24: (new)

Hu et al. discloses a method according to claim 21 wherein the data are supplied in blocks in a block format predetermined by the supplying computer program **(FIG. 6 depicts the relationship among the buffer cache, the TWIP file system, and the TWIP file system device driver. The buffer cache allocates buffer pages for blocks of data on the disk. Each page corresponds to a block on the disk as described in Column 18, lines 16-20.**)

Regarding Claim 25: (new)

Hu et al. discloses a method according to claim 21 wherein the data transmission of the data occurs via a socket connection established between the supplying computer program module and the reading computer program module **(Referring to Figure 4: TCB Tx Queue (812)/Rx Queue (811)--This is the socket queue for transmitting (812) and receiving (811).)**

Regarding Claim 31: (new)

Hu et al. discloses a method to transfer data in a server system comprising at least first and second servers, comprising the steps of:

providing a computer program module**(FIG. 3 describes the software structure for the preferred embodiment for the data-driven multi-processor pipelined model. The functional relationship among the software modules is described at Column 11, line 34)** running on, and supplying data from, the first **(e.g., The Network Interface Card (NIC, 701) receives data from the network as described in Column 11, lines 34-35). FIGS. 11 and 12 are flow charts for data flow from network to storage or vice-versa.)**;

providing a reading computer program module **(In one of Hu's methods e.g., Once a file read request is sent to the host, the TWIP file system does not have to wait for response. It can proceed to process the next connection. After the host acknowledges the request (registration), the TWIP file system will go back to read the file as described in Column 17, lines 13-17)** running on the second server that reads the supplied data; and

selecting one of the following transmission modes

a storage of the data in a file occurs before the reading computer program module reads the data **(e.g., Authorized *non-real-time data transfer* between a network interface and a storage interface described in Column 5, lines 44-45.)**,

a segment-by-segment storage of the data in a file occurs such that the reading computer program module already begins with the reading of a segment while the

supplying computer program module is still supplying data **(e.g., The data content then is divided into segments of integral multiples of a fixed base, a process that we call "base-multiple segmentation" (BMS) technology. For example, a base of y bytes, say 2 Kbytes, is chosen, and all data streams or files are *segmented into chunks* of integral multiples of 2 Kbytes, like 2, 4, or 8 Kbytes (padding it for the last chunk if it is not an exact integral multiple of 2 Kbytes), with an upper limit of, say, 40 Kbytes (20 times y). The maximum size is chosen based-on the requirement of isochronous real-time traffic and the switching speed, such that it will still meet the tightest real-time needs while the switching element serves the largest segments as described in Column 6, lines 3-12)**, and

a transmission of the data between the supplying computer program module and the reading computer program module occurs without buffering **(e.g., Authorized *real-time data transfer* between a network interface and a storage interface as described in Column 5, lines 42-43.)**

Regarding Claim 32: (new)

Hu et al. discloses a computer program system to transfer data in a network of servers, comprising:

a computer program module **(FIG. 3 describes the software structure for the preferred embodiment for the data-driven multi-processor pipelined model. The functional relationship among the software modules is described at Column 11, line 34)** running on the first server and supplying data from the first server **(e.g., The**

Network Interface Card (NIC, 701) receives data from the network as described in Column 11, lines 34-35). FIGS. 11 and 12 are flow charts for data flow from network to storage or vice-versa.);

a reading computer program module that reads the supplied data **(In one of Hu's methods e.g., Once a file read request is sent to the host, the TWIP file system does not have to wait for response. It can proceed to process the next connection. After the host acknowledges the request (registration), the TWIP file system will go back to read the file as described in Column 17, lines 13-17)**; and

the supplying computer program module and the reading computer program module employing one of the following transmission modes

a complete storage of the data in a file occurs before the reading computer program module reads the data **(e.g., Authorized *non-real-time data transfer* between a network interface and a storage interface described in Column 5, lines 44-45.)**,

a segment-by-segment storage of the data in a file occurs such that the reading computer program module already begins with the reading of a segment while the supplying computer program module is still supplying data **(e.g., The data content then is divided into segments of integral multiples of a fixed base, a process that we call "base-multiple segmentation" (BMS) technology. For example, a base of y bytes, say 2 Kbytes, is chosen, and all data streams or files are *segmented into chunks* of integral multiples of 2 Kbytes, like 2, 4, or 8 Kbytes (padding it for the last chunk if it is not an exact integral multiple of 2 Kbytes), with an upper limit of,**

say, 40 Kbytes (20 times y). The maximum size is chosen based-on the requirement of isochronous real-time traffic and the switching speed, such that it will still meet the tightest real-time needs while the switching element serves the largest segments as described in Column 6, lines 3-12), and

a direct transmission of the data between the supplying computer program module and the reading computer program module occurs without buffering **(e.g., Authorized *real-time data transfer* between a network interface and a storage interface as described in Column 5, lines 42-43.)**

Regarding Claim 33: (new)

Hu et al. discloses a computer program system of claim 32 wherein the transmission mode is selected controlled by parameters, and wherein the reading computer program module and the computer program module supplying the data cooperate via the control parameters **(e.g., Once the nature of the traffic is determined, by consulting the Expanded Routing Table (ERT) (with more information than a regular routing table), as shown in FIG. 14, a proper switching path can be selected to forward the traffic with proper QoS measurement. For instance, higher priority traffic can be given more bandwidth and/or lower delay. The forwarded traffic to the network will then be processed with the proper protocol format conversion for transmission with all the necessary error checking and/or correction as described in Column 6, lines 19-27.)**

Regarding Claim 34: (new)

Hu et al. discloses the computer program system of claim 32 wherein in the transmission mode with the direct transmission of the data, the reading computer program module reacts, controlled by the parameters, in one of the following manners when data to be read no longer exists:

the read event is continuously repeated until data to be read is present, or until the reading computer program module receives a notification that data is no longer being supplied, or the read event is aborted **(e.g., In the first method, TWIP sends the filename to the host before TWIP Http engine issues a file read. The host TWIP device driver generates a fake fileopen(Filename) to block any potential host write to the same file. Then the host TWIP device driver sends a write_block_ack back signal back to TWIP. If on the other hand, the host fails to open the file for read, meaning that the host may be writing to the same file and TWIP read request should be held back, no write_block_ack back is to be issued, and the process should retry to open the file later. Once TWIP receives the write_block_ack, TWIP starts reading the file. When TWIP finishes the read, it sends the signal write_block_clear to the host, and the host TWIP device driver then does a fileclose(Filename) as described in Column 16, line 63 - Column 17, line 8.)**

Regarding Claim 35: (new)

Hu et al. discloses the computer program system according to claim 32 wherein the data are supplied in blocks in a block format predetermined by the supplying computer program **(FIG. 6 depicts the relationship among the buffer cache, the TWIP file system, and the TWIP file system device driver. The buffer cache allocates buffer pages for blocks of data on the disk. Each page corresponds to a block on the disk as described in Column 18, lines 16-20.**)

Regarding Claim 36: (new)

Hu et al. discloses the computer program system according to claim 32 wherein the data transmission of the data occurs via a socket connection established between the supplying computer program module and the reading computer program module **(Referring to Figure 4: TCB Tx Queue (812)/Rx Queue (811)--This is the socket queue for transmitting (812) and receiving (811).)**.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. Claims 26-30 and 37-42 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hu et al. as applied to claims 1 and 32 above, and further in view of Nakamura (US 7,262,872).

Regarding Claim 26: (new)

Hu et al. discloses the method according to claim 21.

Hu et al. does not disclose expressly wherein the data to be transferred are print data, and said at least first server comprises a print data server.

Nakamura discloses wherein the data to be transferred are print data, and said at least first server comprises a print data server **(e.g., A print server which accepts a printing request including reference information that indicates a storage location of paper original data from a printing request instruction terminal via a network, designates a printer as a printing destination according to the printing request, and sends a printing instruction to the designated printer, the print server comprising... (As described in Column 19, lines 52-58))**

Nakamura & Hu are combinable because they are from the same field of endeavor; e.g., both references disclose network server systems.

At the time of the invention, it would have been obvious to a person of ordinary skill in the art to use a print data server within a network.

The suggestion/motivation for doing so would be to transmit image data efficiently and to prevent a network from becoming busy as noted by Nakamura in Column 2, lines 31-33.

Therefore, it would have been obvious to combine the print server of Nakamura with the network management system of Hu et al. to obtain the invention as specified in claim 26.

Regarding Claim 27: (new)

Hu et al. further discloses the method according to claim 26 wherein given the storage in segments of data, data of a job are already further processed via the reading computer program module in a subsequent process, while subsequent data of the same job are still being stored **(FIGS. 11 and 12 are flow charts for data flow from network to storage or vice-versa.)**

Regarding Claim 28: (new)

A method according to claim 26 wherein the transmission mode to be applied is established dependent on the job in a job corollary file **(e.g., Once the nature of the traffic is determined, by consulting the Expanded Routing Table (ERT) (with more information than a regular routing table), as shown in FIG. 14, a proper switching path can be selected to forward the traffic with proper QoS measurement. For instance, higher priority traffic can be given more bandwidth and/or lower delay. The forwarded traffic to the network will then be processed with the proper protocol format conversion for transmission with all the necessary error checking and/or correction as described in Column 6, lines 19-27.)**

Regarding Claim 29: (new)

A method according to claim 26 wherein the supplying computer program module runs on the first server, and the reading computer program module runs on a second server **(In this description, there are three types of logical medium interfaces: the network, storage and server(s). In actual implementation, various physical interfaces are possible, e.g., multiple network interfaces or storage interfaces or multiple servers ; There may also be a speed matching function between the network and storage, load balancing functions for servers as described in Column 8, lines 9-20.)**

Regarding Claim 30: (new)

A method according to claim 26 wherein both the supplying computer program module and the reading computer program module run on the first server **(In this description, there are three types of logical medium interfaces: the network, storage and server(s). In actual implementation, various physical interfaces are possible, e.g., multiple network interfaces or storage interfaces or multiple servers ; There may also be a speed matching function between the network and storage, load balancing functions for servers as described in Column 8, lines 9-20.)**

Regarding Claim 37: (new)

Hu et al. discloses the method according to claim 32.

Hu et al. does not disclose expressly wherein the data to be transferred are print data, and said at least first server comprises a print data server.

Nakamura discloses wherein the data to be transferred are print data, and said at least first server comprises a print data server **(e.g., A print server which accepts a printing request including reference information that indicates a storage location of paper original data from a printing request instruction terminal via a network, designates a printer as a printing destination according to the printing request, and sends a printing instruction to the designated printer, the print server comprising... (As described in Column 19, lines 52-58))**

Nakamura & Hu are combinable because they are from the same field of endeavor; e.g., both references disclose network server systems.

At the time of the invention, it would have been obvious to a person of ordinary skill in the art to use a print data server within a network.

The suggestion/motivation for doing so would be to transmit image data efficiently and to prevent a network from becoming busy as noted by Nakamura in Column 2, lines 31-33.

Therefore, it would have been obvious to combine the print server of Nakamura with the network management system of Hu et al. to obtain the invention as specified in claim 37.

Regarding Claim 38: (new)

A computer program system according to claim 37 wherein given the storage in segments of data, data of a job are already further processed via the reading computer program and a subsequent process, while subsequent data of the same job are still being stored **(FIGS. 11 and 12 are flow charts for data flow from network to storage or vice-versa.)**

Regarding Claim 39: (new)

A computer program system according to claim 37 wherein the transmission mode to be applied is established dependent on the job in a job corollary file **(e.g., Once the nature of the traffic is determined, by consulting the Expanded Routing Table (ERT) (with more information than a regular routing table), as shown in FIG. 14, a proper switching path can be selected to forward the traffic with proper QoS measurement. For instance, higher priority traffic can be given more bandwidth and/or lower delay. The forwarded traffic to the network will then be processed with the proper protocol format conversion for transmission with all the necessary error checking and/or correction as described in Column 6, lines 19-27.)**

Regarding Claim 40: (new)

A computer program system according to claim 37 wherein the writing computer program module runs on the first server, and the reading computer program runs on a second server **(In this description, there are three types of logical medium**

interfaces: the network, storage and server(s). In actual implementation, various physical interfaces are possible, e.g., multiple network interfaces or storage interfaces or multiple servers ; There may also be a speed matching function between the network and storage, load balancing functions for servers as described in Column 8, lines 9-20.)

Regarding Claim 41: (new)

A computer program system according to claim 32 wherein both the supplying computer program module and the reading computer program module run on the first server **(In this description, there are three types of logical medium interfaces: the network, storage and server(s). In actual implementation, various physical interfaces are possible, e.g., multiple network interfaces or storage interfaces or multiple servers ; There may also be a speed matching function between the network and storage, load balancing functions for servers as described in Column 8, lines 9-20.)**

Regarding Claim 42: (new)

A computer program system of claim 32 wherein the transmission mode is selected controlled by parameters, and wherein the reading computer program module and the computer program module supplying the data cooperate via the parameters **(e.g., Once the nature of the traffic is determined, by consulting the Expanded Routing Table (ERT) (with more information than a regular routing table), as**

shown in FIG. 14, a proper switching path can be selected to forward the traffic with proper QoS measurement. For instance, higher priority traffic can be given more bandwidth and/or lower delay. The forwarded traffic to the network will then be processed with the proper protocol format conversion for transmission with all the necessary error checking and/or correction as described in Column 6, lines 19-27.)

Conclusion

9. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Cadden et al. (US 6,418,519) discloses a write-behind computer program product is presented which allows writing data to multiple volumes of storage media associated with one or more server nodes in a distributed processing environment.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Neil R. McLean whose telephone number is (571)270-1679. The examiner can normally be reached on Monday through Friday 7:30AM-5:00PM EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David K. Moore can be reached on 571.272.7437. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Neil R. McLean/
Examiner, Art Unit 2625
04/02/2008

/Gabriel I Garcia/

Acting SPE of Art Unit 2625